



HAL
open science

Identification of Key Residues in Proteins Through Centrality Analysis and Flexibility Prediction with RINspector

Guillaume Brysbaert, Théo Mauri, Jerome de Ruyck, Marc Lensink

► **To cite this version:**

Guillaume Brysbaert, Théo Mauri, Jerome de Ruyck, Marc Lensink. Identification of Key Residues in Proteins Through Centrality Analysis and Flexibility Prediction with RINspector. *Current Protocols in Bioinformatics*, 2019, 65 (1), pp.e66. 10.1002/cpbi.66 . hal-02389343

HAL Id: hal-02389343

<https://hal.science/hal-02389343>

Submitted on 10 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Identification of key residues in proteins through centrality analysis and flexibility prediction with RINspecter

Journal:	<i>Current Protocols</i>
Manuscript ID	CP-18-0192
Wiley - Manuscript type:	Protocol
Date Submitted by the Author:	24-Aug-2018
Complete List of Authors:	Brysbart, Guillaume; UGSF, CNRS - Computational biology Mauri, Théo; UGSF, CNRS - Computational biology de Ruyck, Jérôme; UGSF, CNRS - Computational biology Lensink, Marc; UGSF, CNRS - Computational biology
Keywords:	protein structure, residue interaction network, centrality analysis, flexibility prediction
Abstract:	Protein structures inherently contain information as to decipher their functions but the exploitation of this knowledge is not trivial. We recently developed an app for the Cytoscape network visualization and analysis program, called RINspecter, the goal of which was to integrate two different approaches that identify key residues in a protein structure or complex. The first approach consists of calculating centralities on a residue interaction network (RIN) generated from the three-dimensional structure, the second one consists of predicting backbone flexibility and needs only the primary sequence. The identified residues highly correlate with functional relevance and constitute a good set of targets for mutagenesis experiments. Here we present a protocol that details step-by-step how to create a RIN from a structure, and then to calculate centralities and predict flexibilities. We also indicate how to understand and use the results of the analyses.

SCHOLARONE™
Manuscripts



Identification of key residues in proteins through centrality analysis and flexibility prediction with RINspector

Unit Title

Identification of key residues in proteins with RINspector

Guillaume Brysbaert^{1,*}, Théo Mauri¹, Jérôme de Ruyck¹ and Marc F Lensink^{1,*}

CNRS UMR 8576 UGSF, University of Lille, Lille, F-59000, France

E-mail: guillaume.brysbaert@univ-lille.fr; marc.lensink@univ-lille.fr

Tel: +33 3 62 53 17 32

Fax: +33 3 20 43 65 55

Significance Statement

Protein structures are held together by a complex network of covalent and non-covalent interactions. Simplified forms of these networks can be generated from their three-dimensional structure, as solved by e.g. X-ray diffraction, which allows to study these proteins with network analysis tools. These provide fast results that can help in the identification of key amino acids relevant to the function of the protein. In addition, local flexibility predictions provide additional information on flexibility and rigidity of functional regions. Here we present a protocol to use RINspector, a tool which combines network analyses and protein flexibility predictions for the identification of such functionally important residues.

ABSTRACT

Protein structures inherently contain information as to decipher their functions but the exploitation of this knowledge is not trivial. We recently developed an app for the Cytoscape network visualization and analysis program, called RINspector, the goal of which was to integrate two different approaches that identify key residues in a protein structure or complex.

1
2
3 The first approach consists of calculating centralities on a residue interaction network (RIN)
4 generated from the three-dimensional structure, the second one consists of predicting
5 backbone flexibility and needs only the primary sequence. The identified residues highly
6 correlate with functional relevance and constitute a good set of targets for mutagenesis
7 experiments. Here we present a protocol that details step-by-step how to create a RIN from a
8 structure, and then to calculate centralities and predict flexibilities. We also indicate how to
9 understand and use the results of the analyses.
10
11

12 **Keywords: protein structure, residue interaction network, centrality analysis, flexibility**
13 **prediction**
14
15

18 INTRODUCTION

19

20 Understanding the function of a protein in its biological context lies at the heart of fundamental
21 and applied research. Knowledge of the three-dimensional structure of the protein in solution,
22 with or without potential ligands, and in complex with other proteins or not, constitutes a
23 wealth of information to be exploited. These structures can be used for modeling at different
24 levels, like homology modeling, protein-ligand docking, protein-protein docking, dynamic
25 predictions and simulations, or used to identify regions or residues of interest, that may lead to
26 mutagenesis experiments or be used for targeted drug design. We have recently developed a
27 tool called RINspector (Brysbaert et al., 2017, 2018) , the goal of which is to identify key
28 residues in a structure, amenable to mutagenesis experiments. It is included as an app in the
29 Cytoscape network visualization and analysis software (Shannon et al., 2003), accessible
30 through its apps manager. RINspector works on residue interaction networks (RINs) that can be
31 generated from tools like RING2 (Piovesan et al., 2016) or RINerator (Doncheva et al., 2011), or
32 directly inside Cytoscape through a connexion to the Chimera program (Pettersen et al., 2004),
33 which allows visualization and analyses of molecular structures. RINs are network
34 representations of protein structures, where nodes represent residues (or small molecules) and
35 edges represent detected interactions between them. Our tool permits to calculate centralities
36 with a per-node Z-score. We usually consider as relevant nodes with a Z-score ≥ 2 , resulting in a
37 pool of central residues. These residues have been shown to play essential roles in the function,
38 folding or allostery of a protein (e.g. del Sol et al., 2006; Amitai et al., 2004). The RINspector app
39 also enables the user to predict the flexibility of a protein by linking to the Dynamine flexibility
40 prediction server (Cilia et al., 2013, 2014), and to perform in silico mutations for flexibility
41 predictions. The combination of the three functionalities allows to predict the influence of
42 mutation of central residues on local protein flexibility. RINspector has been used in different
43 research projects to help in the identification of important residues in proteins that may be
44 located in their core or their surface, alone or in complex (Brysbaert et al., 2017; Laulumaa et
45 al., 2018; de Ruyck et al., 2018).
46
47
48
49
50
51

52 Here we present a protocol to use RINspector in Cytoscape that covers the creation of a RIN
53 from Chimera, through use of the structureViz2 Cytoscape app (Morris et al., 2007),
54 modifications of the RIN, centrality analyses and backbone flexibility predictions. We also
55
56
57
58
59
60

1
2
3 present a support protocol for the execution of the same steps through an external script that
4 we provide in Python3 and R.
5
6
7

8 ***BASIC PROTOCOL 1***

9 **Creating a Residue Interaction Network with Cytoscape, structureViz2 and Chimera**

10
11 Residue interaction networks are networks generated from a PDB structure, where nodes are
12 residues or small molecules and edges are detected interactions. They can be imported into
13 Cytoscape like any network, but an easy way to generate them for Cytoscape is to use the
14 structureViz2 app, which links to the Chimera visualization program. In addition, this
15 connection permits that any residue or node selected in the one representation is also selected
16 in the other (structure/Chimera or network/Cytoscape). We will generate the RIN for the
17 structure of the catalytic domain of the PARP-1 enzyme (PDB ID: 4R6E). The Poly [ADP-ribose]
18 polymerase 1 transfers ADP-ribose groups to target proteins and is involved in the regulation of
19 many cellular processes like cell division, transcriptional regulation or DNA damage response
20 (Gibson and Kraus, 2012). Mutation or dysregulation of the enzyme has been associated to
21 certain diseases including cancer. Specific inhibitors targeting PARP-1 have been designed with
22 the objective to inhibit cancer progression. Several structure files for the PARP-1 catalytic
23 domain are available in the PDB databank. We chose 4R6E because validation criteria are
24 correct and the structure shows no missing residues (Thorsell et al., 2017). The protocol
25 explains how to start the tools, open a structure and generate a RIN from it.
26
27
28
29
30
31
32
33

34 ***Necessary resources***

35 *Hardware*

36
37
38 Computer with Internet connection, under Windows, Mac or Linux and at least 8 GB
39 RAM.
40

41 *Software*

42
43 Java SE runtime environment 8

44
45 Cytoscape 3.6.1 (or higher) from <http://www.cytoscape.org/>

46
47 Chimera 1.13 (or higher) from <https://www.cgl.ucsf.edu/chimera/>
48
49

- 50
51 1. Install and configure the structureViz2 and RINalyzer apps
52
53
54
55
56
57
58
59
60

1
2
3 Launch Cytoscape, access the Apps menu in the top bar, then App manager (Figure 1).
4 Search for “structureViz2”, click “Install”. Then search for “RINalyzer” (Doncheva et al.,
5 2011) and do the same.
6

7 The complete path to the UCSF Chimera application may be required (depending on the
8 installation), before it can be run from Cytoscape: more information about the
9 configuration can be found on the structureViz2 tutorial webpage
10 <http://www.cgl.ucsf.edu/cytoscape/structureViz2/tutorial.html>. In particular the section
11 1.g) about the path configuration to Chimera:
12

13 “Depending on your installation, it might be necessary to tell structureViz2 exactly where
14 the Chimera binary is. This is done by adding the path in the structureViz Settings...
15 (Apps→structureViz→Settings...). Note that on the Macintosh this should be the path to
16 the Chimera executable, not the Chimera.app directory. If you install Chimera in your
17 Applications folder (the recommended install location), then set the path to:
18 /Applications/Chimera.app/Contents/MacOS/. Once you have entered the path, click OK.
19 This will “remember” this setting for the current session and Cytoscape installation so
20 that you will not have to set it again unless you have a new installation of UCSF Chimera
21 or a session file that uses another instance of Chimera.”
22
23
24

25 2. Load the PDB structure in Chimera

26
27 In App/StructureViz, launch Chimera. If the path was well specified in the settings, it
28 starts the Chimera program and links it to Cytoscape through structureViz2. A
29 connection window entitled “Cytoscape Molecule Structure Navigator” opens in the
30 background, which also permits to generate a RIN. In Chimera, select File/Fetch by ID. In
31 the PDB part, type “4R6E” and Fetch (Figure 2). Alternatively, download the file from a
32 PDB website like RCSB <https://www.rcsb.org/structure/4R6E> and load the file in Chimera
33 using File/Open. The structure file contains 4 copies of the catalytic domain of PARP-1,
34 consisting of ~400 amino acids with the niraparib inhibitor in its catalytic site. The
35 connection window shows the list of residues of each chain and the inhibitor molecules
36 (Figure 3).
37
38
39
40
41

42 3. Create a RIN

43
44 The “Cytoscape Molecule Structure Navigator” connection window allows interactions
45 with Chimera. The 4 chains all represent the same catalytic domain and differ only by a
46 few residues at their terminal ends; we focus on chain A. Click on “4R6E chain A (404
47 residues)” in the connection window. Launch Residue Network Generation in the
48 Chimera menu. It opens a window, specific to the RIN generation (Figure 4). For more
49 information about the RIN generation, see the structureViz2 website
50 (<http://www.cgl.ucsf.edu/cytoscape/structureViz2/>), the RINalyzer website
51 (<http://rinalyzer.de/tut/tutorial10.php>) and the Chimera user’s guide
52 (<https://www.cgl.ucsf.edu/chimera/docs/UsersGuide/>).
53
54
55
56
57
58
59
60

1
2
3 **Parameters:** *The links between the nodes in the network will depend on the selected*
4 *parameters. In this tutorial, we want to focus on contacts, thereby considering the RIN*
5 *as a contact map. With the exception of some specific cases, contact map analyses are*
6 *usually sufficient to give good results. Consequently, the “Include contacts” option*
7 *should be selected. Default parameters for contact detection are recommended and*
8 *there is no need to add C-alpha distance contacts (leave “Include distances between CA*
9 *atoms” unticked). Do not add hydrogens since they are not present in the X-ray structure*
10 *and do not detect hydrogen bonds, to stick to contact detection only. Thus untick “Add*
11 *hydrogens” and “Include hydrogen bonds”. “Include clashes” should be unselected as*
12 *well. In this example, we do not consider the backbone connectivity, in order to focus on*
13 *the folding properties of the domain and backbone-independent inter-amino acid*
14 *interactions. The “Ignore water” option can be left unselected. The inclusion of water in*
15 *the network usually leads to a larger set of central residues but those obtained without*
16 *water are usually included in this larger set. The “Include combined edges option” won’t*
17 *have any effect on the analysis and can be unselected to simplify the network (less*
18 *edges). Once all parameters are set, click validate to create the RIN (Figure 5).*
19 *The RINalyzer app displays the nodes following the orientation of the structure in*
20 *Chimera, colors the nodes in function of secondary structure and the edges in function of*
21 *the type of contact (see “Style” menu in Cytoscape for association of colors). Each node*
22 *selected in Cytoscape automatically selects the corresponding residue (or small*
23 *molecule) in Chimera and vice-versa. Colors can be synchronized in*
24 *App/StructureViz/Synchronize Residue Colors. It is recommended to save the Cytoscape*
25 *session at this point (File/Save as).*

32 33 34 35 **BASIC PROTOCOL 2**

36 37 **Performing centrality analyses of a RIN with RINspector**

38
39
40 Once a RIN is opened in Cytoscape, it can be analyzed with network centrality tools. These tools
41 are Cytoscape core network analysis tools or apps like CytoNCA, Centiscape or more specialized
42 ones for RINs like RINalyzer or RINspector. RINspector permits to easily identify specific central
43 residues by automatically calculating a Z-score for each of them and highlighting in the network
44 view only those with relevant Z-scores ($Z\text{-score} \geq 2$). The centrality calculation proposed by (del
45 Sol et al., 2006) is implemented in RINspector as well as more conventional betweenness and
46 closeness centrality calculations, to which the calculation of Z-scores is added. In addition to
47 this, the DynaMine flexibility prediction server can be queried to retrieve the flexibility
48 prediction of a protein chain, after which the user can perform in silico mutagenesis of the
49 protein sequence in order to visualize the impact of the mutation on the predicted flexibility of
50 the protein chain. In this protocol, we describe how to perform centrality analyses of the 4R6E
51 RIN generated in the first protocol, how to predict flexibility and how to perform in silico
52 mutations for flexibility predictions.
53
54
55
56
57
58
59
60

Necessary resources

Hardware

Computer with Internet connection, under Windows, Mac or Linux and at least 8 GB RAM.

Software

Java SE runtime environment 8

Cytoscape 3.6.1 (or higher) from <http://www.cytoscape.org/>

Chimera 1.13 (or higher) from <https://www.cgl.ucsf.edu/chimera/>

1. Install RINspector and prepare the RIN for further analyses

In Cytoscape, open the Apps/App manager (Figure 1), search and install “RINspector”. If not already displayed, open a Cytoscape session which contains a RIN for the 4R6E chain A structure. In this structure, the catalytic site of PARP-1 presents a pocket in which the niraparib inhibitor is located. The inhibitor is in the center of the structure and gathers all the centralities of the network if conserved. This is indicative of the optimized interactions surrounding the inhibitor, but little informative for our purpose. Consequently, we are going to clone the RIN and remove the niraparib ligand (named 3JD), as well as SO4 and GOL present in the structure. Select File/New/Network/Clone current network. Rename the collection then the network, adding “_without_ligands” in the name. Select this new network in the control panel on the left of the Cytoscape window. In the search bar at the top (at the right of menu icons), type 3JD GOL SO4 +Enter to select these three ligands (Figure 6). Finally, remove these nodes in Edit/Delete selected Nodes and Edges (note that they are removed only from the network, not from the structure in Chimera).

2. Run a Residue Centrality Analysis (RCA)

In Apps/RINspector, run a Centrality Analysis. Three types of centrality analyses are available in the current release 1.1.0; a description of these is available in (Brysbaert et al., 2017). Some explanation is also provided in the Help option of Apps/RINspector. The default selection is the Residue Centrality Analysis (RCA) which consists of the calculation of the change in average shortest path length of the network (also called characteristic path length) under removal of individual nodes. The shortest path between two nodes is the path in the network required to connect the first node with the second one with the minimum number of edges. The shortest path length is this number of edges. The average shortest path length is the average of shortest path lengths of all possible pairs

of nodes in a network. The more the average shortest path length increases upon deletion of a node, the higher its Z-score. In case of high Z-scores, it means that the node (residue/small molecule) may play an important role in the communication inside the network (structure).

Tip: If the structure that corresponds to the RIN is opened in Chimera through structureViz2, the connection between Cytoscape and Chimera makes the running much slower. To reduce this calculation time, we advise to close Chimera in Apps/StructureViz/Exit Chimera before running the RCA centrality. Chimera and the structure can be loaded again once the calculation has finished.

Run a RCA, selecting it in Apps/RINspecter/Centrality analysis. After a few minutes, the analyzed network appears with a new style (Figure 7). Only Z-scores ≥ 2 are colored, from yellow (2) to red (≥ 4). The bigger the node size and labels, the bigger the Z-score value. A "Z-score_RCA" style is created in the "Style" tab at the top of the control panel (above the list of networks). The calculated Z-score of each node appears in a new column in the Node Table entitled "Z-score_RCA". If Chimera has been closed before running a RCA, open it again from Apps/StructureViz/Launch Chimera, then fetch the 4R6E structure (cf Basic Protocol 1).

3. Run a Betweenness Centrality Analysis (BCA) and a Closeness Centrality Analysis (CCA)

Similar as for RCA, run a BCA then a CCA (in Apps/RINspecter/Centrality analysis). Both considering shortest paths, BCA evaluates how often a node is crossed by paths between all node pairs while CCA evaluates the distance of a node to all other nodes in the network. For BCA and CCA, it is not necessary to exit Chimera because the calculation times are very short. Once a calculation has finished, the "Z-score_BCA" and "Z-score_CCA" styles are created, as well as a column for each that contains the Z-score per node in the node table. It is possible to switch from one representation of the network to another one, changing the style in the Style tab of the control panel on the left of the Cytoscape window. Switching between the "Z-score_RCA", "Z-score_BCA" and "Z-score_CCA" styles, we see that the sets of central residues between RCA and BCA globally overlap while the CCA analysis gives very few results. This is a general observation due to the fact that the closeness values are more homogeneously distributed and consequently few Z-scores are significant. Should the user be interested in this calculation, we recommend to use directly the closeness values instead of the Z-scores. They can be retrieved with Cytoscape core analyses tools (Tools/NetworkAnalyzer), RINalyzer, CytoNCA or Centiscape apps.

4. Synchronize colors with the structure in Chimera

Once centralities calculated, colors of the RIN can be synchronized with the structure. Select the "Z-score_RCA" style in the Style tab of the control panel in Cytoscape. In

1
2
3 *Chimera, 4 chains are available, we only want to keep chain A that was used to generate*
4 *the RIN.*

- 5
6 - *hide chains B, C and D: in top menus Select/Chain/A, Select/Invert (all models), then*
7 *Actions/Ribbon/Hide and Actions/[Atoms/Bonds]/Hide.*
8 - *hide the GOL and SO4: Select/Residue/GOL and Actions/[Atoms/Bonds]/Hide. Idem*
9 *for SO4.*
10 - *activate the automatic selection between Cytoscape and Chimera views: in*
11 *Cytoscape, Select/Nodes/Select all nodes.*
12
13

14 *All the residues of chain A are selected in Cytoscape and Chimera. The ligand is not*
15 *selected because it is not present in the network. We are going to draw the chain in*
16 *ribbon, except the residues that have a Z-score ≥ 2 and the 3JD ligand.*

- 17
18 - *In Chimera, Actions/ [Atoms/Bonds]/Hide.*
19 - *In Cytoscape, go to "Select" in the top tabs of the control panel (Figure 8), click on*
20 *"+" /Column Filter, choose "Z-score_RCA" (like the current style) and filter between 2*
21 *and the maximum value that is set automatically. Click on "Apply" at the bottom of*
22 *the panel.*
23 - *In Chimera, select Actions/[Atoms/Bonds]/Show.*
24
25

26 *Finally, synchronize colors*

- 27 - *In Cytoscape, select Apps/StructureViz/Synchronize residue colors, and validate.*
28 *Residue labels can be displayed in Chimera with Actions/Label/Residue/One-letter*
29 *code + specifier and white colored in Action/Color/all options, select first "residue*
30 *labels" in the right part of the pop-up window, then click "white" and close the colors*
31 *window. Lastly unselect all.*
32
33

34 *After a reorientation of the structure, you should get a view like Figure 9A. Nodes in the*
35 *Cytoscape network can be repositioned automatically thanks to the RINLayout available*
36 *in the Layout menu, which should give Figure 9B.*
37
38
39
40
41
42

43 **BASIC PROTOCOL 3**

44 **Performing DynaMine flexibility predictions of a RIN/protein with RINspecter**

45
46
47 The backbone flexibility of a protein depends on its amino acid composition and has a major
48 effect on function. Rigid parts of a structure will be less susceptible to conformation changes
49 than flexible ones. Even inside catalytic pockets, depending on the flexibility of the protein in
50 these clefts, a ligand may bind more or less effectively. Tools such as DynaMine, accessible via a
51 dedicated webserver, give a prediction of the flexibility of a protein from its sequence;
52 DynaMine calculates NMR S^2 order parameters for each residue of the protein sequence. The
53 higher the S^2 value, the more rigid the protein is predicted to be at this location. DynaMine can
54 be called from RINspecter, submitting the sequence associated to a RIN. This protocol explains
55
56
57
58
59
60

1
2
3 how to run DynaMine for a RIN and how to predict the backbone flexibility of the same protein
4 sequence after in silico mutation of individual residues. Coupled with the previously described
5 centrality analysis, one can focus on the flexibility of central residues.
6
7
8
9

10 **Necessary resources**

11 *Hardware*

12
13
14 Computer with Internet connection, under Windows, Mac or Linux and at least 8 GB
15 RAM.
16

17 *Software*

18
19 Java SE runtime environment 8

20
21 Cytoscape 3.6.1 (or higher) from <http://www.cytoscape.org/>

22
23 Chimera 1.13 (or higher) from <https://www.cgl.ucsf.edu/chimera/>
24
25
26

- 27 1. Predict the backbone flexibility of the protein

28
29
30 *If not installed, install the RINspector app (cf Basic Protocol 2). A RIN like the one*
31 *generated in Basic Protocol 1 has to be opened in Cytoscape with Z-scores calculated like*
32 *in Basic Protocol 2. In Apps/RINspector, select “DynaMine Prediction”. We advise the*
33 *user to always use uninterrupted structures to build a RIN to avoid losing nodes and*
34 *edges. For DynaMine, holes in a RIN are a problem because the sequence submitted to*
35 *the server for flexibility prediction is built from the RIN. Therefore, the missing residues*
36 *won’t be visible in the resulting flexibility graph but the prediction will be biased.*
37
38

- 39 2. Visualize results

40
41
42 *The prediction run displays a graph in the result panel in Cytoscape and adapts the style*
43 *of the network (Style entitled “DynaMine_S2”). We are going to select the central*
44 *residues found by RCA in Basic Protocol 2: select the nodes with $Z\text{-score}_{RCA} \geq 2$ as*
45 *described in Basic Protocol 2 – part 4 (Figure 10).*

46
47 *The result panel shows a flexibility prediction graph with S^2 on the Y axis and residues*
48 *listed in sequence order on the X axis (if not displayed, the result panel can be shown in*
49 *View/Show Results Panel). A S^2 value is associated to each residue, listed in the*
50 *dedicated S^2 column of the Node table. This S^2 value can be interpreted as a rigidity*
51 *factor: the higher the value, the lower is the predicted flexibility at this location,*
52 *considering ranges 0 to 0.69 as flexible, 0.69 to 0.8 as context dependent and superior to*
53 *0.8 as rigid (see (Cilia et al., 2014, 2013)).*
54
55
56
57
58
59
60

1
2
3 *Each residue selected in the network is also selected in the graph, shown as a red bar.*
4 *The graph can be zoomed in with left click-and-slide and zoomed out with right click-*
5 *and-slide. A right click on the plot gives access to additional options like export and*
6 *properties. The selected residues are listed in a dedicated table in the result panel.*
7 *The colors of the network can be synchronized with the structure in*
8 *Apps/StructureViz/Synchronize Residue Colors (Figure 11).*
9
10

11 3. Perform in-silico mutations for flexibility predictions

12
13
14 *In the DynaMine result panel (Figure 12), a mutation can be set for any selected residue*
15 *in the sequence by simply specifying the one-letter code of the residues to mutate in the*
16 *“Residue mutation” column. For instance, Lys893, which has a relevant Z-score in RCA*
17 *and BCA was shown to be critical for the catalytic activity of PARP1 (Simonin et al.,*
18 *1990): when mutated to Ile, PARP1 loses its catalytic activity. In the DynaMine result*
19 *panel, type I in the residue table next to K893 +Enter. Then run DynaMine again and*
20 *zoom in at the area around K893 (zoom box). A new prediction is queried to the*
21 *DynaMine server with the mutated sequence. A random color is set for the new curve*
22 *(red in Figure 12). This prediction shows that the variant is predicted to rigidify the*
23 *backbone around the residue 893 compared to the wild type. In addition to obvious*
24 *electrostatic and hydrophobic properties of the residue, the rigidification of the*
25 *backbone could have an impact on the interaction with the ligands.*
26 *Several mutations can be performed at a same time. All prediction results can be*
27 *downloaded with the “Download results” buttons.*
28
29
30
31
32
33
34
35

36 **SUPPORT PROTOCOL 1**

37 **Performing centralities calculations and flexibility predictions from a script in Python or R**

38
39
40 The basic protocols above are descriptive of a step-by-step usage of the RINspector tool. It
41 becomes a bit laborious when several structures have to be analyzed sequentially like in the
42 case of NMR data for instance. Since its release 3.3, Cytoscape embeds a tool named CyREST
43 (Ono et al., 2015) that enables the user to call functions of Cytoscape from an external program
44 written in Python or R, for instance. RINspector in its release 1.1.0 permits such a call of its
45 centrality analyses and DynaMine prediction functions (Brysbaert et al., 2018). This protocol
46 describes how to call RINspector functions from a script in Python or R to perform calculations
47 on the 4 chains of the PDB 4R6E.
48
49
50
51
52

53 **Necessary resources**

54 *Hardware*

Computer with Internet connection, under Windows, Mac or Linux and at least 8 GB RAM.

Software

Java SE runtime environment 8

Cytoscape 3.6.1 (or higher) from <http://www.cytoscape.org/>

Chimera 1.13 (or higher) from <https://www.cgl.ucsf.edu/chimera/>

Python 3.5.2 minimum or R 3.2.3

1. Open Cytoscape and structures in Chimera

If not installed, install and configure Cytoscape apps StructureViz2, RINalyzer and RINspector like described in Basic Protocol 1. Then launch Chimera from StructureViz2. Fetch the PDB ID 4R6E.

2. Create one RIN per chain with a script

All the available commands that can be interrogated through the REST service available in Cytoscape are described in the menus Help/Automation/CyREST API and Help/Automation/CyREST Command API (RINspector commands are described in this menu). To create one RIN per chain from a PDB opened in Chimera, with the same parameters as in the Basic Protocol 1, each chain has to be selected and the URL <http://localhost:1234/v1/commands/structureViz/createRIN> has to be queried with the HTTP POST method in JSON format. In the following script, this action is performed for each chain and the ligands are removed.

Create an empty Python 3 file entitled "create_RINs.py" and paste the following code:

```
#required packages
import requests, time, string

#parameters
listOfChains = ["A", "B", "C", "D"]
pdb = "4R6E"
listOfLigands = ["GOL", "SO4", "3JD"]
urlSelect = "http://localhost:1234/v1/commands/structureViz/send?command=select%20%3A."
urlCreateRIN = "http://localhost:1234/v1/commands/structureViz/createRIN?"
RINparameters = "includeHBonds=false&ignoreWater=false"
urlDelete = "http://localhost:1234/v1/commands/network/delete"

#generates one RIN per chain
for i in listOfChains:

    #selects the chain and creates a RIN
    requests.get(urlSelect+i)
    time.sleep(5)
```

```

requests.get(urlCreateRIN+RINparameters+"&networkName="+pdb+"_"+i)

#deletes the ligands
for l in listOfLigands:
  requests.post(urlDelete, json={"nodeList": "ResType:"+l})

```

or in a "create_RINs.R" file:

```

#required packages
require("httr")
require("jsonlite")

#parameters
listOfChains <- c("A","B","C","D")
pdb <- "4R6E"
listOfLigands <- c("GOL","SO4","3JD")
urlSelect <- "http://localhost:1234/v1/commands/structureViz/send?command=select%20%3A."
urlCreateRIN <- "http://localhost:1234/v1/commands/structureViz/createRIN?"
RINparameters <- "includeHBonds=false&ignoreWater=false"
urlDelete <- "http://localhost:1234/v1/commands/network/delete"

#generates one RIN per chain
for(i in listOfChains)
{
  #selects the chain and creates a RIN
  chainSelection <- paste(c(urlSelect, i),collapse="")
  httr::GET(chainSelection)
  Sys.sleep(5)
  rinCreation <- paste(c(urlCreateRIN, RINparameters, "&networkName=", pdb, "_", i), collapse = "")
  createNetwork <- httr::GET(rinCreation)

  #deletes the ligands
  for (l in listOfLigands)
  {
    ligandName <- paste(c("ResType:",l), collapse = "")
    deleteLigand <- POST(url = urlDelete, body = list(nodeList=ligandName), encode="json")
  }
}

```

Then run the script while Cytoscape and Chimera are running, with the structure opened in Chimera: in a terminal, reach the location of the script and type "python3 create_RINs.py" or "R -q --vanilla < create_RINs.R". The script creates thus one RIN per chain.

3. Perform a RCA for each RIN

To run a centrality analysis on a network, the URL <http://localhost:1234/v1/commands/rinspector/centrality> has to be queried with the HTTP POST method, specifying the type of centrality in parameters among "Residue (ASPL change under removal of individual nodes) - RCA", "Betweenness - BCA" and "Closeness - CCA". In the following script, we only calculate RCA. The same script can be used to run BCA or CCA on the same RINs, simply modifying the "typeOfCentrality" parameter with the desired centrality listed before. In an empty Python 3 "RCA_all.py" file, write:

```

#required packages

```

```

1
2
3 import requests, json, os, sys
4
5 #parameters
6 urlList = "http://localhost:1234/v1/commands/network/list"
7 urlSetNetwork = "http://localhost:1234/v1/commands/network/set%20current"
8 urlCentrality = "http://localhost:1234/v1/commands/rinspector/centrality"
9 typeOfCentrality = "Residue (ASPL change under removal of individual nodes) - RCA"
10
11 #gets the list of networks SUIDs
12 networksList = requests.post(urlList)
13 networksList = networksList.json()
14 networksList = (networksList["data"]["networks"])
15
16 #calculates centralities for each network
17 for i in networksList:
18
19     #sets the network
20     requests.post(urlSetNetwork, json={"network": "SUID:"+str(i)})
21
22     #calculates centralities
23     calcul = requests.post(urlCentrality, json={"analysisType": typeOfCentrality})

```

Or in a "RCA_all.R" file:

```

24 #required packages
25 require("httr")
26 require("jsonlite")
27
28 #parameters
29 urlList <- "http://localhost:1234/v1/commands/network/list"
30 urlSetNetwork <- "http://localhost:1234/v1/commands/network/set%20current"
31 urlCentrality <- "http://localhost:1234/v1/commands/rinspector/centrality"
32 typeOfCentrality <- "Residue (ASPL change under removal of individual nodes) - RCA"
33
34 #function that sends a POST request in JSON format and returns the content of the result
35 requestPOST <- function(urlString, bodyList, encodeString)
36 {
37     request <- POST(url=urlString, body=bodyList, encode=encodeString)
38     resultFromJSON <- unname(fromJSON(rawToChar(request$content)))
39     return(resultFromJSON[[1]])
40 }
41
42 #gets the list of networks SUIDs
43 networksList <- requestPOST(urlString=urlList, bodyList=NULL, encodeString="json")
44
45 #calculates centralities for each network
46 for(i in 1:length(networksList$networks))
47 {
48     #sets the network
49     networkID <- paste("SUID:",networksList$networks[i],sep="")
50     requestPOST(urlString=urlSetNetwork, bodyList=list(network=networkID), encodeString="json")
51
52     #calculates centralities
53     requestPOST(urlString=urlCentrality, bodyList=list(analysisType=typeOfCentrality), encode="json")
54 }

```

Then run the script while the RINs are opened in Cytoscape, it will calculate RCA for all the RINs opened. As mentioned before, it is recommended to exit Chimera before executing the RCA to reduce calculation times (Apps/StructureViz/Exit Chimera). To run the script, in a terminal, reach its location and type: "python3 RCA_all.py" or "R -q --vanilla < RCA_all.R".

4. Perform a DynaMine prediction for each RIN

For DynaMine, the URL `http://localhost:1234/v1/commands/rinspector/dynamine` has to be queried with the HTTP POST method after selection of a RIN and chain. In the same way as centrality calculations, in a “flexibility_all.py” Python file, write:

```
#required packages
import requests, json, os, sys

#parameters
urlList = "http://localhost:1234/v1/commands/network/list"
urlSetNetwork = "http://localhost:1234/v1/commands/network/set%20current"
urlGetNetwork = "http://localhost:1234/v1/commands/network/get"
urlGetColumn = "http://localhost:1234/v1/commands/table/get%20column"
urlDynamine = "http://localhost:1234/v1/commands/rinspector/dynamine"

#gets the list of networks SUIDs
listOfNetworks = requests.post(urlList)
listOfNetworks = listOfNetworks.json()
listOfNetworks = (listOfNetworks["data"]["networks"])

#calculates DynaMine flexibility predictions for the only chain of each network
for i in listOfNetworks:

    #sets the network
    requests.post(urlSetNetwork, json={"network":"SUID:"+str(i)})

    #gets the network name
    req = requests.post(urlGetNetwork, json={"network":"current"})
    networkJSON = req.json()
    networkName = networkJSON["data"]["name"]

    #gets the first chain ID (only one chain here)
    req = requests.post(urlGetColumn, json={"table":networkName+" default node", "column":"ResChain"})
    chainJSON = (req.json())
    chain = chainJSON["data"]["values"][1]

    #predicts flexibility of the chain with DynaMine
    requests.post(urlDynamine, json = {"chainToAnalyze":chain})
```

Or in a “flexibility_all.R” file:

```
#required packages
require("httr")
require("jsonlite")

#parameters
urlList <- "http://localhost:1234/v1/commands/network/list"
urlSetNetwork <- "http://localhost:1234/v1/commands/network/set%20current"
urlGetNetwork <- "http://localhost:1234/v1/commands/network/get"
urlGetColumn <- "http://localhost:1234/v1/commands/table/get%20column"
urlDynamine <- "http://localhost:1234/v1/commands/rinspector/dynamine"

#function that sends a POST request in JSON format and returns the content of the result
requestPOST <- function(urlString, bodyList, encodeString)
{
    request <- POST(url=urlString, body=bodyList, encode=encodeString)
    resultFromJSON <- unname(fromJSON(rawToChar(request$content)))
    return(resultFromJSON[[1]])
}
```

```

1
2
3
4 #gets the list of networks SUIDs
5 networkList <- requestPOST(urlString=urlList, bodyList=NULL, encodeString="json")
6
7 #calculates DynaMine flexibility predictions for the only chain of each network
8 for(i in 1:length(networkList$networks))
9 {
10     #sets the network
11     networkID <- paste("SUID:", networkList$networks[i],sep="")
12     requestPOST(urlString=urlSetNetwork, bodyList=list(network=networkID), encodeString="json")
13
14     #gets the network name
15     req <- requestPOST(urlString=urlGetNetwork, bodyList=list(network="current"), encode="json")
16     networkName <- req$name
17
18     #gets the first chain ID (only one chain here)
19     tableAndColumn <- list(table=paste(networkName, "default node"), column="ResChain")
20     req <- requestPOST(urlString=urlGetColumn, bodyList=tableAndColumn, encodeString="json")
21     chain <- req$values[1]
22
23     #predicts flexibility of the chain with DynaMine
24     requestPOST(urlString=urlDynamine, bodyList=list(chainToAnalyze=chain), encodeString="json")
25 }

```

Then run the script while the RINs are opened in Cytoscape: in a terminal, reach its location and type: “python3 flexibility_all.py” or “R -q --vanilla < flexibility_all.R”. The flexibility prediction of each chain is thus performed. More information about automation and advanced scripts for RINspector is available in (Brysbart et al., 2018).

COMMENTARY

Background Information

Methods to identify crucial residues in a PDB structure are very welcome to improve the understanding of the protein’s function. These methods are usually based on the structure itself and based on energy calculations like alanine scanning or molecular dynamic simulations, and sometimes require long running times. Residue interaction networks are generated from three-dimensional structures but their intrinsic properties permit to perform analyses that are complementary to any of the methods listed above. These analyses are network based and relatively fast to run. They allow in particular to identify residues that were shown to be important for the function of proteins, e.g (Amitai et al., 2004; del Sol et al., 2006). In combination with other approaches centrality analyses can help in the selection of residues that constitute targets for mutagenesis experiments or inhibitor design. (de Ruyck et al., 2018) is an example of integration of different methods for the identification of residues essential for the binding of two protein domains. The goal of RINspector was to design a tool that can quickly identify central residues. By integrating three different types of centrality calculations with backbone flexibility prediction the tool allows the user to identify those residues that are important for the function or the structural integrity of a protein and investigate the impact of mutagenesis experiments on local flexibility.

Critical Parameters

Only the choice of centrality is a parameter that determines the resulting scores but it is nearly costless to run the three of them because BCA and CCA calculations are very fast. The RCA is the most time-consuming but usually does not exceed a few minutes.

Troubleshooting

When the RIN contains a lot of nodes (i.e. the structure contains a lot of residues) or the sum of the nodes of all the RINs for which to run centrality analyses is high and RCA has to be performed, the RAM can be filled during calculations because a lot of temporary networks are generated. We advise to use a computer with at least 16Gb RAM, the higher, the more comfortable to run calculations. Also for RCA, we recommend to run them directly from a freshly opened session of Cytoscape. Temporary networks that may have been generated before would eventually be purged by the Java garbage collector but this release of memory may take some time. For RINspector, BCA can be run first because this calculation is the least time and memory consuming.

When a RCA is executed, it is recommended to exit Chimera if opened because the synchronisation with Cytoscape slows down the process considerably. The structure can be reopened after the run is over.

Fetching a PDB structure or querying the DynaMine server relies on a functional internet connection. The sequence that is sent for DynaMine requests is built from the RIN, which means that there should be no missing residues to get a relevant flexibility prediction.

In case of running scripts like described in Support Protocol 1, the developer has sometimes to consider a short delay for synchronization and selection of a structure. That is why the *sleep* function is called between the selection of a structure and the generation of the RIN.

Statistical Analyses

BCA and RCA results partially overlap. Depending on the objective of the work, it can be interesting to consider the union of these results or the intersection. CCA results could also be integrated but usually give few nodes with Z-score ≥ 2 . If the user is interested in this measure, the best may be to calculate directly the closeness values from Cytoscape core network analysis tools or apps like RINalyzer, CytoNCA or Centiscape.

For some networks, specifically disconnected ones, the results may vary more between RCA and BCA because of the nature of the calculation: while the betweenness relates to how much a node is crossed by shortest paths, the RCA relates to the impact of the deletion of a node on the average shortest path length of the network. BCA is usually more sensitive to disconnections in networks.

1
2
3 In the protocol, we removed the ligands from the structures because due to its nature and
4 location, the inhibitor molecule accumulates all weight of the network, as can be seen by a
5 simple run of RCA or BCA on the raw RIN (before removing ligands). It is evident that in cases
6 where the ligand is not so central, or to visualize the impact of the ligand on centralities, it can
7 be conserved for calculations.
8
9

10 11 12 **Understanding Results** 13

14 The centrality analyses fully rely on the links between the nodes in the networks, which means
15 that the results will directly depend on the parameter set for the generation of the RINs.
16 Typically, considering only contacts already gives satisfactory results, which is what we
17 presented in this protocol.
18
19

20 According to the RIN, BCA and RCA can give fewer or more central residues. If a subselection is
21 required for mutageneses experiments for instance, the intersection of results of both methods
22 can be considered. DynaMine flexibility prediction can also be used to select residues that
23 belong to a more or less flexible region. In silico mutagenesis can be performed to predict
24 flexibility of the variant and choose which type of mutation to perform in wet lab.
25
26

27 In addition, centrality analyses of other structures of the same protein(s) can be run and results
28 considered for intersection to reduce the subset. These runs can be performed automatically
29 through a script like those described in Support Protocol 1. In the given example, the set of
30 residues with Z-scores ≥ 2 in BCA and RCA in the 4 structures is Asn767, Arg865, Phe891, Ile994,
31 Arg878 and Lys893. Lys893 for instance was shown to be critical for PARP1 catalytic activity,
32 which was lost upon mutation to Ile (Simonin et al., 1990). When considering the set of those
33 that are relevant in BCA and RCA in at least three structures, Cys908 appears, which was shown
34 to reduce the PARP1 enzymatic activity to less than 0.5% if mutated to Arg (Rolli et al., 1997).
35 When considering only central nodes in BCA in the four structures, Asn868 emerges, the
36 mutation of which to Ser was found to reduce the catalytic activity to 4%. Phe869 appears as
37 well, which upon mutation to Tyr was found to reduce the catalytic activity to 25%.
38
39
40

41 Thus, multiple sets of central residues coming from multiple structures can be considered.
42 When coupled to backbone flexibility predictions and considering the positioning of these
43 central residues in the structure, they help in the selection of crucial amino acids for
44 mutagenesis experiments or drug design.
45
46
47
48

49 **Time Considerations** 50

51 BCA and CCA calculations last a few seconds while running RCA shouldn't take more than a few
52 minutes. Querying the DynaMine server takes a few seconds.
53

54 Running the 4 protocols can be done within 2 hours if all necessary resources are
55 present/installed.
56
57
58
59
60

LITERATURE CITED

- Amitai, G., Shemesh, A., Sitbon, E., Shklar, M., Netanel, D., Venger, I., and Pietrokovski, S. 2004. Network analysis of protein structures identifies functional residues. *J Mol Biol* 344:1135–46.
- Brysbaert, G., Lorgouilloux, K., Vranken, W., and Lensink, M. F. 2017. RINspecter: a Cytoscape app for centrality analyses and DynaMine flexibility prediction. *Bioinformatics (Oxford, England)*.
- Brysbaert, G., Mauri, T., and Lensink, M. F. 2018. Comparing protein structures with RINspecter automation in Cytoscape. *F1000Research* 7:563.
- Cilia, E., Pancsa, R., Tompa, P., Lenaerts, T., and Vranken, W. F. 2013. From protein sequence to dynamics and disorder with DynaMine. *Nat Commun* 4:2741.
- Cilia, E., Pancsa, R., Tompa, P., Lenaerts, T., and Vranken, W. F. 2014. The DynaMine webserver: predicting protein dynamics from sequence. *Nucleic Acids Res* 42:W264-70.
- Doncheva, N. T., Klein, K., Domingues, F. S., and Albrecht, M. 2011. Analyzing and visualizing residue networks of protein structures. *Trends Biochem Sci* 36:179–82.
- Gibson, B. A., and Kraus, W. L. 2012. New insights into the molecular and cellular functions of poly(ADP-ribose) and PARPs. *Nature Reviews. Molecular Cell Biology* 13:411–424.
- Laulumaa, S., Nieminen, T., Raasakka, A., Krokengen, O. C., Safaryan, A., Hallin, E. I., Brysbaert, G., Lensink, M. F., Ruskamo, S., Vattulainen, I., et al. 2018. Structure and dynamics of a human myelin protein P2 portal region mutant indicate opening of the β barrel in fatty acid binding proteins. *BMC structural biology* 18:8.
- Morris, J. H., Huang, C. C., Babbitt, P. C., and Ferrin, T. E. 2007. structureViz: linking Cytoscape and UCSF Chimera. *Bioinformatics* 23:2345–7.
- Ono, K., Muetze, T., Kolishovski, G., Shannon, P., and Demchak, B. 2015. CyREST: Turbocharging Cytoscape Access for External Tools via a RESTful API. *F1000Research* 4:478.
- Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., and Ferrin, T. E. 2004. UCSF Chimera--a visualization system for exploratory research and analysis. *J Comput Chem* 25:1605–12.
- Piovesan, D., Minervini, G., and Tosatto, S. C. E. 2016. The RING 2.0 web server for high quality residue interaction networks. *Nucleic Acids Research* 44:W367-374.

- 1
2
3 Rolli, V., O'Farrell, M., Ménissier-de Murcia, J., and de Murcia, G. 1997. Random mutagenesis of
4 the poly(ADP-ribose) polymerase catalytic domain reveals amino acids involved in
5 polymer branching. *Biochemistry* 36:12147–12154.
6
7
8 de Ruyck, J., Brysbaert, G., Villeret, V., Aumercier, M., and Lensink, M. F. 2018. Computational
9 characterization of the binding mode between oncoprotein Ets-1 and DNA-repair
10 enzymes. *Proteins*.
11
12 Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski,
13 B., and Ideker, T. 2003. Cytoscape: a software environment for integrated models of
14 biomolecular interaction networks. *Genome Res* 13:2498–504.
15
16
17 Simonin, F., Ménissier-de Murcia, J., Poch, O., Muller, S., Gradwohl, G., Molinete, M., Penning,
18 C., Keith, G., and de Murcia, G. 1990. Expression and site-directed mutagenesis of the
19 catalytic domain of human poly(ADP-ribose)polymerase in *Escherichia coli*. Lysine 893 is
20 critical for activity. *The Journal of Biological Chemistry* 265:19249–19256.
21
22
23 del Sol, A., Fujihashi, H., Amoros, D., and Nussinov, R. 2006. Residues crucial for maintaining
24 short paths in network communication mediate signaling in proteins. *Molecular Systems*
25 *Biology* 2:2006.0019.
26
27
28 Thorsell, A.-G., Ekblad, T., Karlberg, T., Löw, M., Pinto, A. F., Trésaugues, L., Moche, M., Cohen,
29 M. S., and Schüler, H. 2017. Structural Basis for Potency and Promiscuity in Poly(ADP-
30 ribose) Polymerase (PARP) and Tankyrase Inhibitors. *Journal of Medicinal Chemistry*
31 60:1262–1271.
32
33
34
35

36 INTERNET RESOURCES

37
38 <http://www.cytoscape.org/>

39
40 *Cytoscape*

41
42 <http://apps.cytoscape.org/>

43
44 *Cytoscape app store for RINspector, structureViz2, RINalyzer and other apps with description*
45 *and resources*

46
47 <https://www.cgl.ucsf.edu/chimera/>

48
49 *Chimera*

50
51 <http://dynamine.ibsquare.be/>

52
53
54 *DynaMine server*
55
56
57
58
59
60

FIGURE LEGENDS

Figure 1 – App Manager inside Cytoscape that allows to find and install apps.

Figure 2 – Opening of the PARP1 catalytic domain with niraparib inhibitor (4R6E) in Chimera. A - Dedicated window in Chimera to fetch a structure; the PDB ID is typed inside the textbox; B - 3D view of the structure, which contains 4 chains (A to D).

Figure 3 – Cytoscape Molecular Structure Navigator window that bridges between Chimera and Cytoscape; the Chimera menu permits in particular to generate Residue Interaction Networks (RINs) after selection of one or several chain(s).

Figure 4 – Residue Interaction Network (RIN) generation dialog that enables to generate a RIN, specifying parameters.

Figure 5 – Cytoscape window with the general menu at the top, Control Panel on the left, Node, Edge and Network Table at the bottom and network view on the right. A Residue Interaction Network (RIN) was generated from the 4R6E PDB structure. Nodes are colored in function of their secondary structure (red - helix; grey - loop; blue - sheet); edges are colored in shades of blue depending on the type of interaction (contact main chain – main chain, contact main chain - side chain, contact main chain – water...). The position of the nodes depends on the coordinates of residues in the 3D structure from which the RIN has been created.

Figure 6 – Same Residue Interaction Network (RIN) as figure 5 with ligands 3JD, GOL and SO4 selected (yellow) for deletion. The ligands are searched in the top search bar.

Figure 7 – Residue Centrality Analysis (RCA) performed on the generated RIN of 4R6E (cf figures 5 and 6). Nodes are colored in function of their Z-score_{RCA}, from yellow (Z-score = 2) to red (Z-score ≥ 4), other nodes are white, node size and label size also depend on the Z-score: the higher, the bigger.

Figure 8 – Selection tab of the control panel in Cytoscape. Here, nodes with Z-scores_{RCA} ≥ 2 are selected.

Figure 9 – Interplay between Chimera (A) and Cytoscape (B). A – Structure of 4R6E, chain A; backbone is white colored, only amino acids with Z-score_{RCA} ≥ 2 show their atoms in stick representation (hydrogens excepted) and their color depends on the Z-score_{RCA} value: from yellow (Z-score = 2) to red (Z-score ≥ 4). B – RIN representation of the 4R6E structure; nodes colors are the same as for (A), they are synchronized, their size and label size also depend on the Z-score: the higher, the bigger. The position of the nodes depends on the coordinates of residues in the 3D structure (A).

Figure 10 – DynaMine flexibility prediction of the backbone of 4R6E – chain A; for each residue, a S^2 value is calculated, the higher the value, the more rigid the backbone is predicted to be at the location of the amino acid. Middle - Nodes of the Residue Interaction Network (RIN) are colored in function of the S^2 value: red (0 to 0.69), green (0.69 to 0.8) and blue (≥ 0.8); selected nodes are in yellow. Right – DynaMine result panel which shows predicted flexibility graph and

1
2
3 selected nodes/residues in a dedicated table; the selected nodes/residues are also shown as
4 red bars in the plot (more details in figure 12).
5

6
7 **Figure 11** – Structure of 4R6E – chain A colored in function of DynaMine flexibility prediction
8 scores. Colors are synchronized with the Residue Interaction Network (RIN) (Figure 10) and
9 depend on the DynaMine S^2 values: red (0 to 0.69), green (0.69 to 0.8) and blue (≥ 0.8).
10

11 **Figure 12** – DynaMine result panel inside Cytoscape. Bottom left – Table with selected residues,
12 the residue K893 is mutated to I; Bottom right – Buttons to run DynaMine after mutations have
13 been set into the left-hand side table and buttons for export of data and clearance of graph;
14 Top – Flexibility prediction graphs zoomed in at the K893 region; residues are displayed on the
15 X axis, S^2 on the Y axis; the green curve corresponds to the flexibility predictions of the wild
16 type and the red one to the K893I mutant; selected residues are shown with red bars.
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

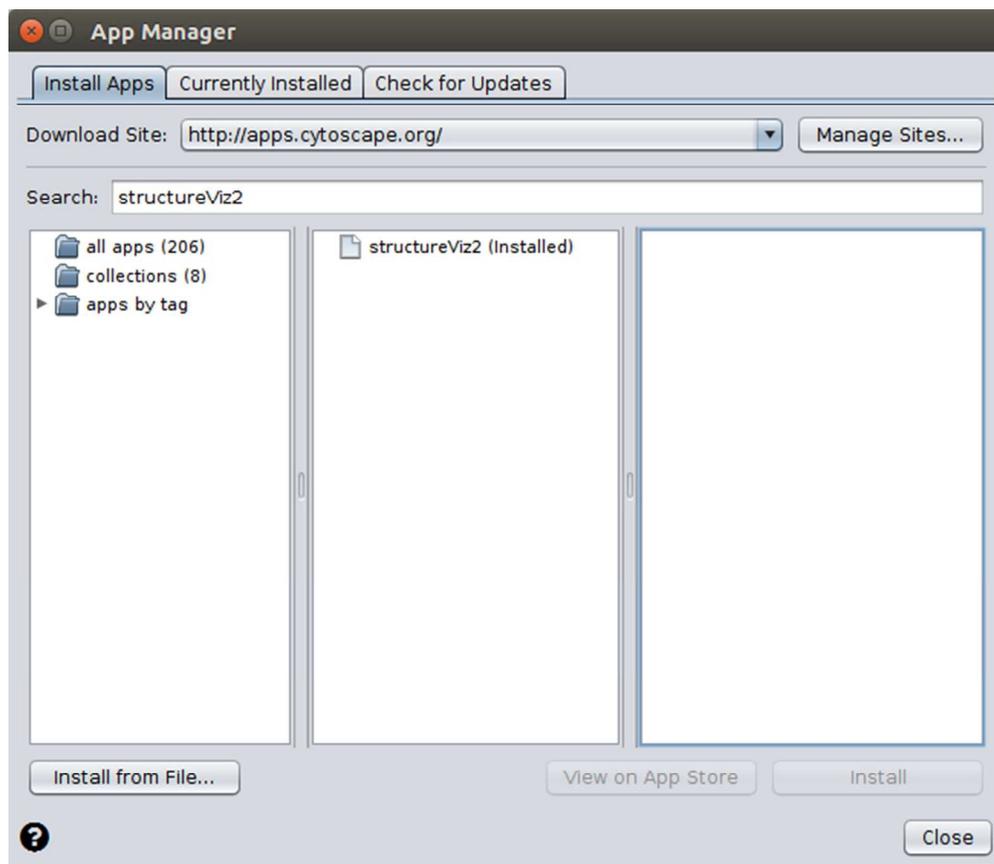


Figure 1 - App Manager inside Cytoscape that allows to find and install apps.

55x47mm (300 x 300 DPI)

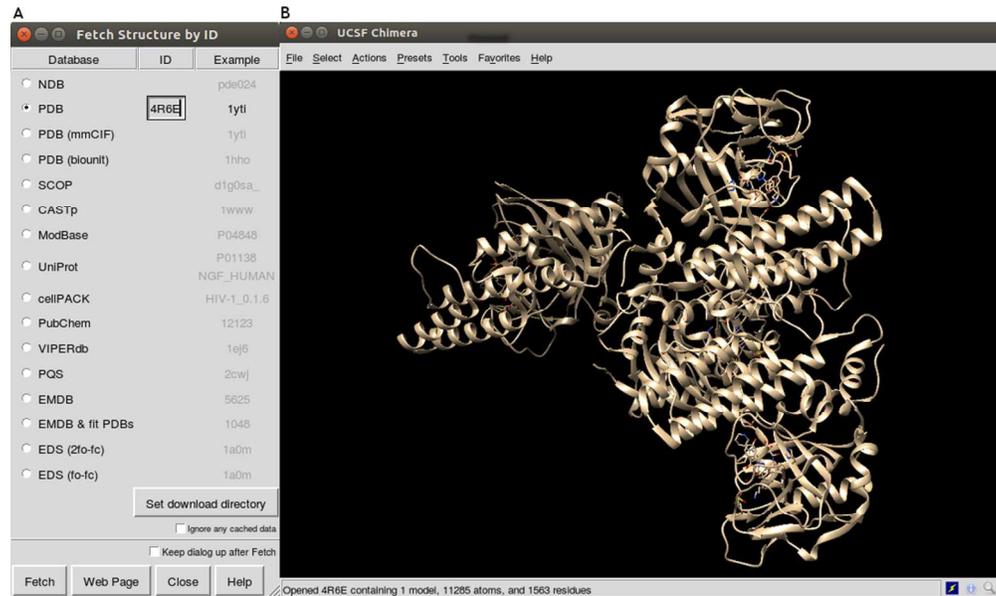
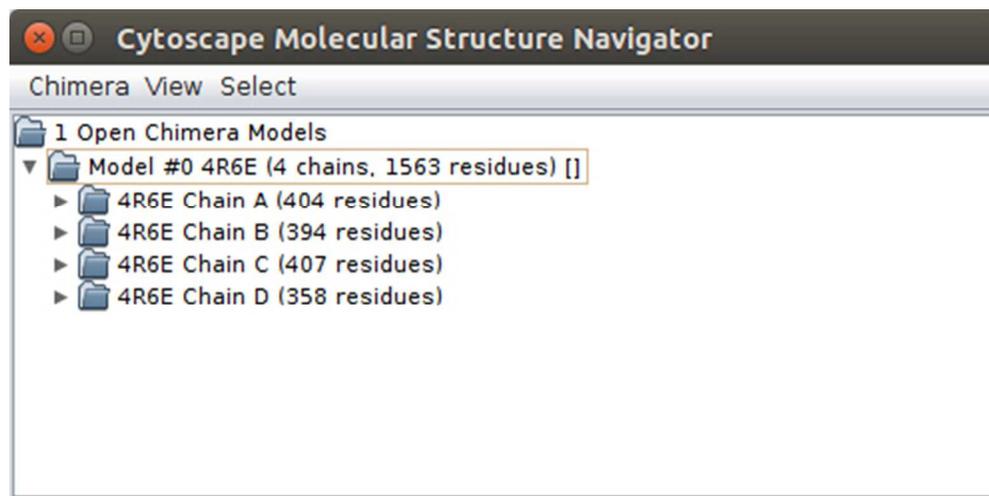


Figure 2 - Opening of the PARP1 catalytic domain with niraparib inhibitor (4R6E) in Chimera. A - Dedicated window in Chimera to fetch a structure; the PDB ID is typed inside the textbox; B - 3D view of the structure, which contains 4 chains (A to D).

93x57mm (300 x 300 DPI)



23 Figure 3 - Cytoscape Molecular Structure Navigator window that bridges between Chimera and
24 Cytoscape; the Chimera menu permits in particular to generate Residue Interaction Networks
25 (RINs) after selection of one or several chain(s).
26

27 42x20mm (300 x 300 DPI)
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

Residue Interaction Network Generation Dialog

General

Name of new network: RIN_4R6E_A

Include interactions: Within selection

Add hydrogens:

Ignore water:

Include combined edges:

Contacts

Include contacts:

Overlap cutoff (in angstroms): -4E-1

HBond allowance (in angstroms): 0E0

Bond separation: 4

Clashes

Include clashes:

Overlap cutoff (in angstroms): 0,6

HBond allowance (in angstroms): 0,4

Bond separation: 4

Hydrogen bonds

Include hydrogen bonds:

Remove redundant contacts:

Add tolerances to strict criteria:

Distance tolerance (in angstroms): 0,4

Angle tolerance (in degrees): 20

Connectivity

Include (backbone) connectivity:

Distance

Include distances between CA atoms:

Distance cutoff (in angstroms): 5

OK Cancel

45 Figure 4 - Residue Interaction Network (RIN) generation dialog that enables to generate a RIN,
46 specifying parameters.

47
48 49x75mm (300 x 300 DPI)

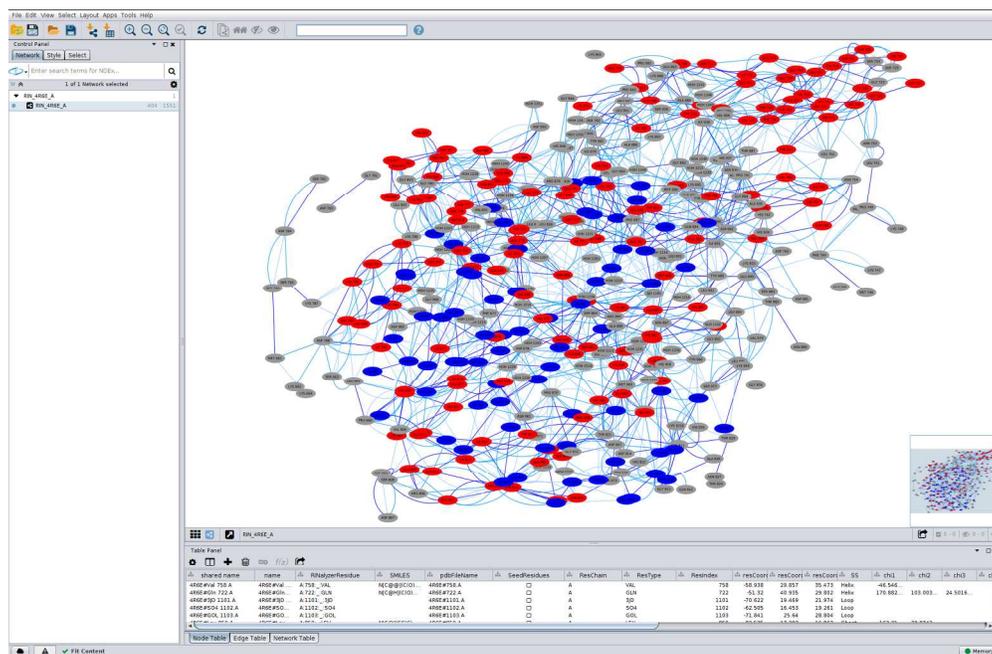


Figure 5 - Cytoscape window with the general menu at the top, Control Panel on the left, Node, Edge and Network Table at the bottom and network view on the right. A Residue Interaction Network (RIN) was generated from the 4R6E PDB structure. Nodes are colored in function of their secondary structure (red - helix; grey - loop; blue - sheet); edges are colored in shades of blue depending on the type of interaction (contact main chain – main chain, contact main chain - side chain, contact main chain – water...). The position of the nodes depends on the coordinates of residues in the 3D structure from which the RIN has been created.

181x117mm (300 x 300 DPI)

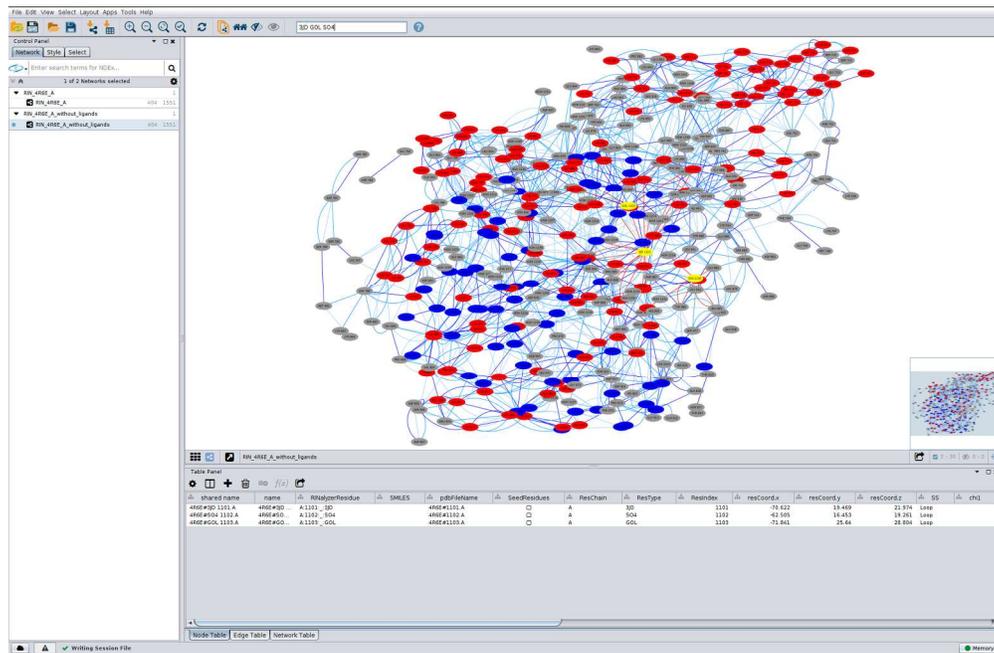


Figure 6 - Same Residue Interaction Network (RIN) as figure 5 with ligands 3JD, GOL and SO4 selected (yellow) for deletion. The ligands are searched in the top search bar.

181x117mm (300 x 300 DPI)

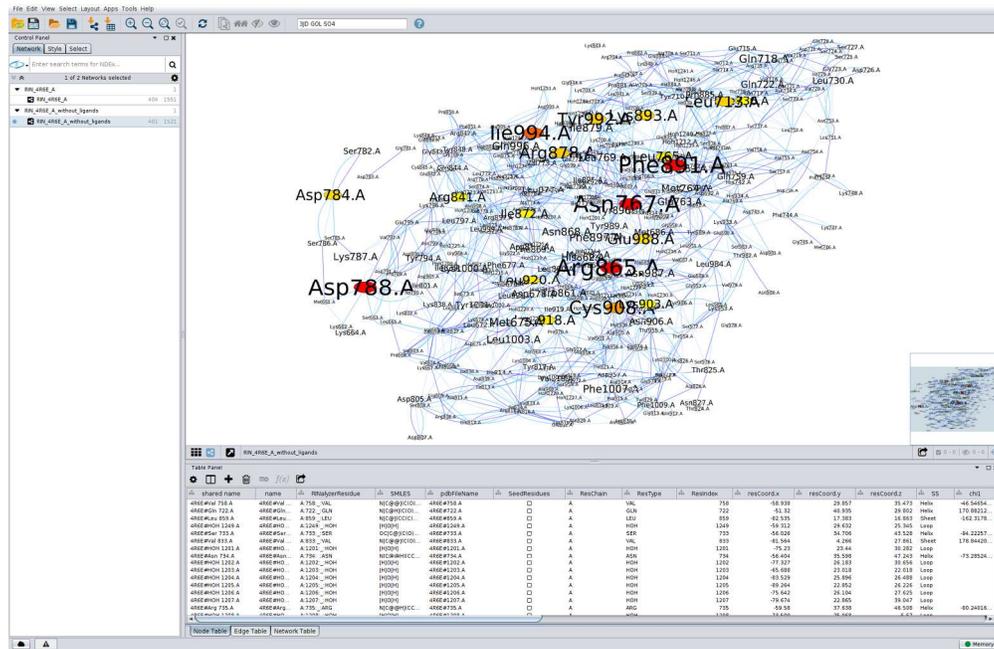


Figure 7 - Residue Centrality Analysis (RCA) performed on the generated RIN of 4R6E (cf figures 5 and 6). Nodes are colored in function of their Z-score_RCA, from yellow (Z-score = 2) to red (Z-score ≥ 4), other nodes are white, node size and label size also depend on the Z-score: the higher, the bigger.

181x117mm (300 x 300 DPI)

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

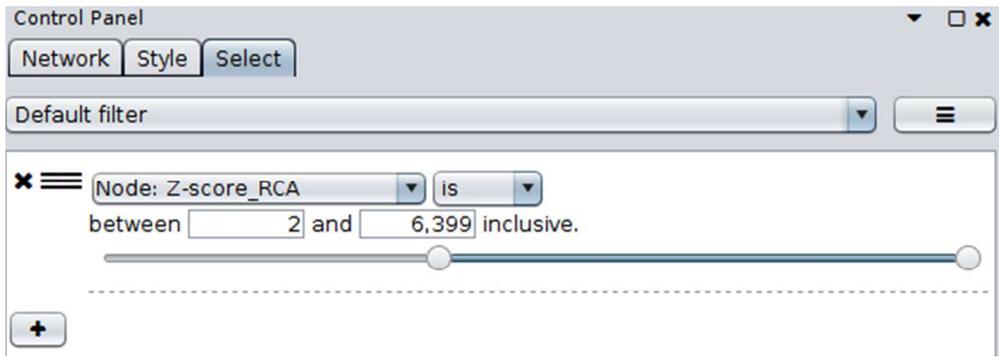


Figure 8 - Selection tab of the control panel in Cytoscape. Here, nodes with Z-scores_RCA ≥ 2 are selected.

49x17mm (300 x 300 DPI)

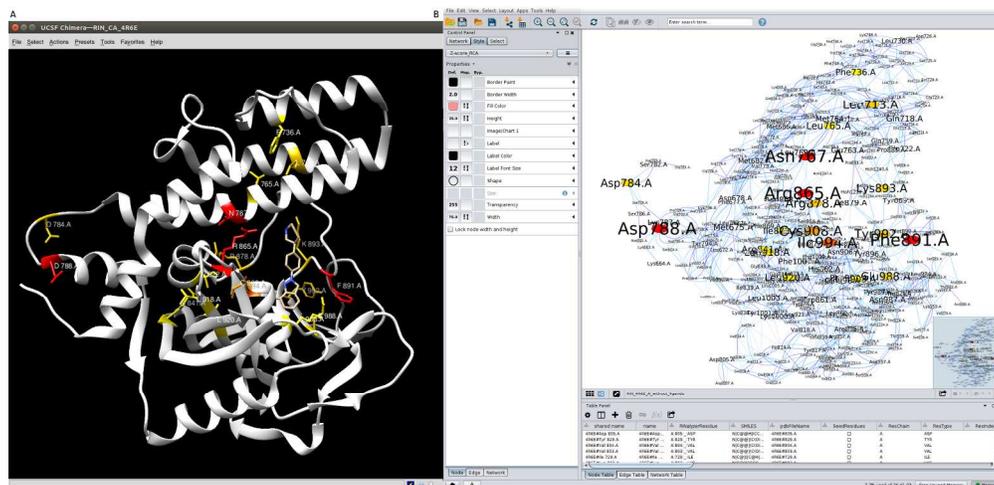


Figure 9 - Interplay between Chimera (A) and Cytoscape (B). A – Structure of 4R6E, chain A; backbone is white colored, only amino acids with $Z\text{-score_RCA} \geq 2$ show their atoms in stick representation (hydrogens excepted) and their color depends on the $Z\text{-score_RCA}$ value: from yellow ($Z\text{-score} = 2$) to red ($Z\text{-score} \geq 4$). B – RIN representation of the 4R6E structure; nodes colors are the same as for (A), they are synchronized, their size and label size also depend on the $Z\text{-score}$: the higher, the bigger. The position of the nodes depends on the coordinates of residues in the 3D structure (A).

189x92mm (300 x 300 DPI)

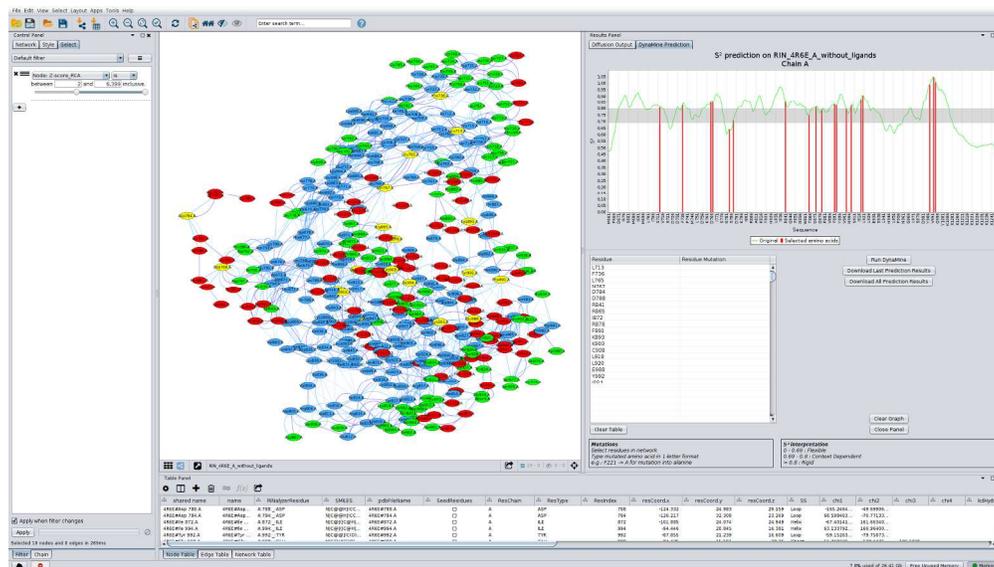


Figure 10 - DynaMine flexibility prediction of the backbone of 4R6E – chain A; for each residue, a S² value is calculated, the higher the value, the more rigid the backbone is predicted to be at the location of the amino acid. Middle - Nodes of the Residue Interaction Network (RIN) are colored in function of the S² value: red (0 to 0.69), green (0.69 to 0.8) and blue (≥ 0.8); selected nodes are in yellow. Right - DynaMine result panel which shows predicted flexibility graph and selected nodes/residues in a dedicated table; the selected nodes/residues are also shown as red bars in the plot (more details in figure 12).

211x119mm (300 x 300 DPI)

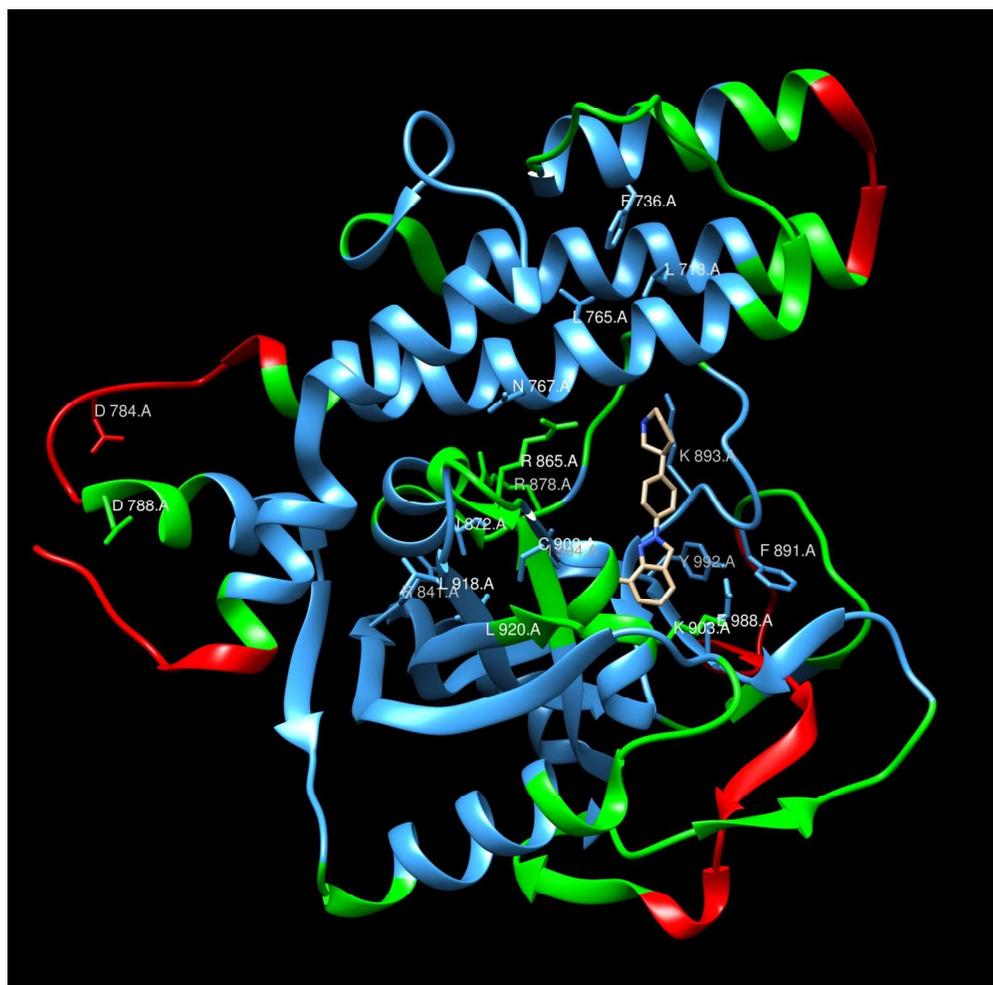


Figure 11 - Structure of 4R6E – chain A colored in function of DynaMine flexibility prediction scores. Colors are synchronized with the Residue Interaction Network (RIN) (Figure 10) and depend on the DynaMine S^2 values: red (0 to 0.69), green (0.69 to 0.8) and blue (≥ 0.8).

127x125mm (300 x 300 DPI)

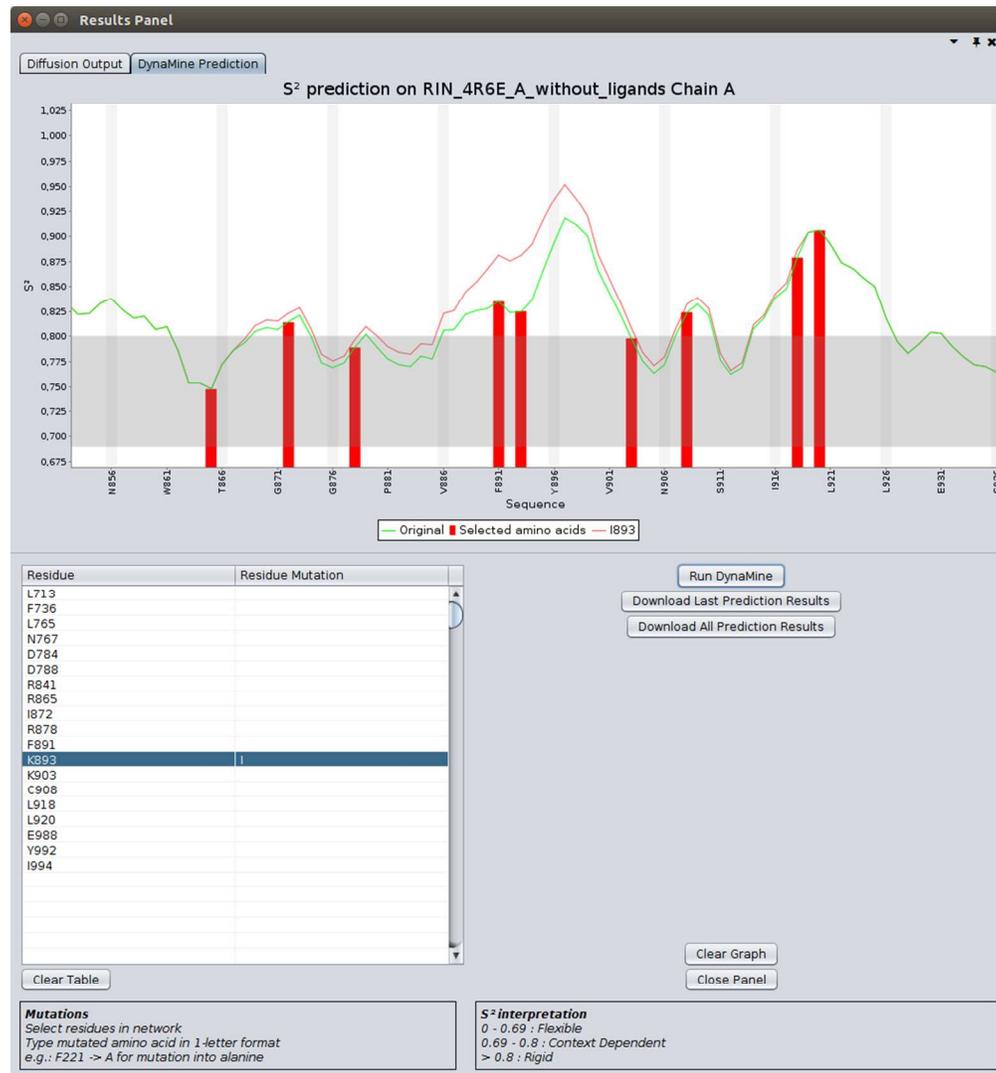


Figure 12 - DynaMine result panel inside Cytoscape. Bottom left – Table with selected residues, the residue K893 is mutated to I; Bottom right – Buttons to run DynaMine after mutations have been set into the left-hand side table and buttons for export of data and clearance of graph; Top – Flexibility prediction graphs zoomed in at the K893 region; residues are displayed on the X axis, S² on the Y axis; the green curve corresponds to the flexibility predictions of the wild type and the red one to the K893I mutant; selected residues are shown with red bars.

89x96mm (300 x 300 DPI)